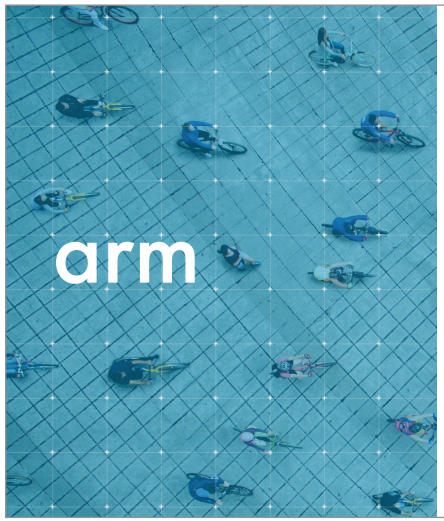
Cortex-A55 POP IP on 16FFC: Is Performance still “King”?

Arm

ABSTRACT

New Cortex-A CPU cores push us to new performance targets and we gain performance with each new process node. We love to see increases in gigahertz. But more and more, we hear that performance is not King. Cost is the new ruler. More designs are focusing on area and power, highly emphasizing the importance of die cost. This is true across market segments, not just for chips used in IoT applications. This presentation will talk about how Arm Coretx-A55 POP IP on TSMC16FFC not only focuses on a performance boost, but puts much more effort into area and power optimization. We will show the implementation results including area, power and performance for the latest Cortex-A55 CPU using Arm TSMC16FFC POP IP. We will show how we did timing and DRC fixing on this 16nm FinFET process, we will highlight features that benefit the implementation results, and we will share lessons learned that can further improve the SoC design.



Cortex-A55 POP IP on 16FFC: Is Performance still “King”?

JC Yu, Technical Marketing Manager
Jaspreet Singh, Design Engineer

TSMC OIP, San Jose
13th Sep 2017

Arm POP IP on 16nm addresses needs of designers

Need to shorten the design cycle	POP IP is a comprehensive, fully validated Cortex-A CPU implementation solution Includes Physical IP, floorplans and reference implementation scripts
Need to lower technical and schedule risk	POP IP is developed and tuned in synergy with RTL over several iterations All Physical IP and implementation issues have been identified and solved by EAC date
Need to achieve market-leading PPA	POP undergoes extensive iterative floorplan exploration and design tuning to deliver market- leading PPA Our record in 16nm FinFET technology is a testament to the hard work behind POP development

But is market-leading PPA all about performance?

Each generation of Cortex-A CPU cores push us to new performance targets

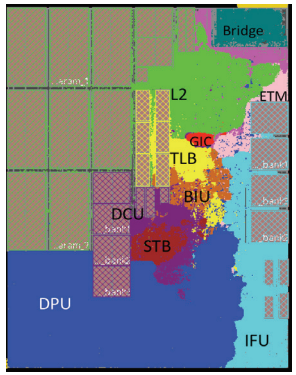
- But more and more, we hear that performance is not King
- More designs are focusing on area and power
- The importance of die cost is dominating

Arm Cortex-A55 POP IP on TSMC 16FFC

- An excellent example of the changing PPA focus to power and area
- Puts more implementation effort into area and power optimization, not just a performance boost

Cortex-A55 POP IP configuration details

Design Configuration
Process : TSMC 16nm FFC 11 Layer +AP w/ routing on M2-M9
L1 cache : 32K
L2 cache : 256K
ECC : Enabled
Crypto : Enabled
L2 Latency: Input=1 and Output=2
Power Domains : 2 per CPU with LS and ELS required
PVT Corners : Setup: RC Max: TT/1.0v/85c Power: TT/0.8v/85c Hold: RC Min: FFGNP/1.05v/125c
Physical IP Configuration
9-track BASE/HPK/PMK : ULC20 for data , ULC16 for clock
Metal Stack : 11m_2xa1xd3xe2y2r_utrdl



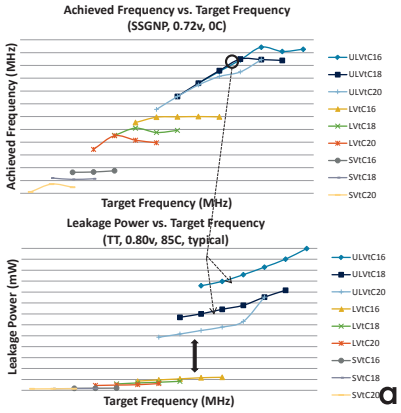
Identify appropriate Vt/CL choices with synthesis shmoo

Target low power with Vt/CL choices

Synthesis shmoo is generated to understand performance & leakage trends with different Vt/CL

Helps make the right choices for implementation before the implementation

The shmoo analysis is designed to resolve optimization strategy for Vt/CL performance and power optimization



Understand dynamic power tradeoffs

Make the right tradeoffs for your design needs: POP IP does it for you

	Performance worst case corner	Performance typical corner	Area	Leakage Power typical corner	Dynamic Power typical corner	Total Power typical corner
SVt	1.0x	1.0x	1.0x	1.0x	1.0x	1.0x
LVt	1.49x	1.30x	1.0x	2.73x	1.30x	1.69x
ULVt	2.13x	1.70x	1.0x	28.33x	1.47x	2.95x

Cortex-A55 POP IP optimizes your design for dynamic power as well as leakage power

Data based on Cortex-A53 single-core implementation with SC9MC 9 track multi-channel libraries on 16nm

Drive area down from synthesis

‘Mixed-Vt’ Shmoo to evaluate performance/power tradeoffs

Physical-aware synthesis

- Use with a ‘tuned’ floorplan to improve correlation between synthesis and placement stages

Selectively Ungrouping hierarchies for better optimization

Path group definition and weight exploration

- Addition of cost groups in synthesis to enable better optimization of paths

Tweak Margining strategy at synthesis stage to achieve correlation with P&R

- Higher Uncertainty @Synthesis followed by sharp fall off @ placement stage

Cell profiling at synthesis stage to understand the top-k Cells being picked from the library

- Correlate cell usage with logic depth of critical paths. Ex: Check for AOI/NAND ratio
- Highly dependent upon the EDA tool & synthesis flow being deployed

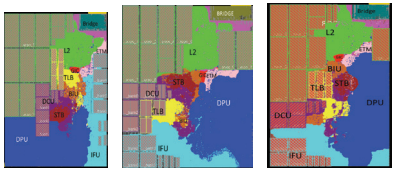
  arm

Optimal floorplanning for low area and power

Fix timing paths that are floorplan related

Multiple trials based on design partitioning

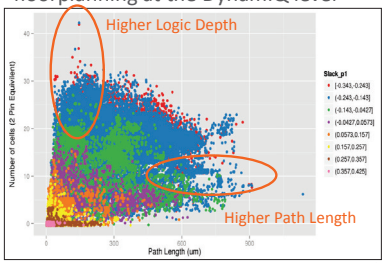
Look at different aspect ratios - “X * Y” options for CPU



Floorplan Topologies for Cortex-A55 CPU

Macro placement trials tune PPA at early stages

Different CPU topologies to ease floorplanning at the DynamIQ level



  arm

Arm Artisan Power Grid Architect

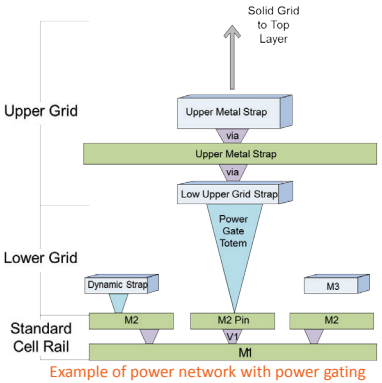
Arm Power Grid Architect (PGA) automates critical aspects of floorplanning

- Extremely critical for FinFET and double pattern designs
- Useful also for complex power grids in 28nm

Considers Arm recommended structures and Arm Artisan Physical IP

- Insert power grids and power gates
- Stitch together power gate sleep signals
- Insert boundary finishing cells as needed
- Detect orphan row or placement site violations

PGA runs on industry standard floorplanning tools



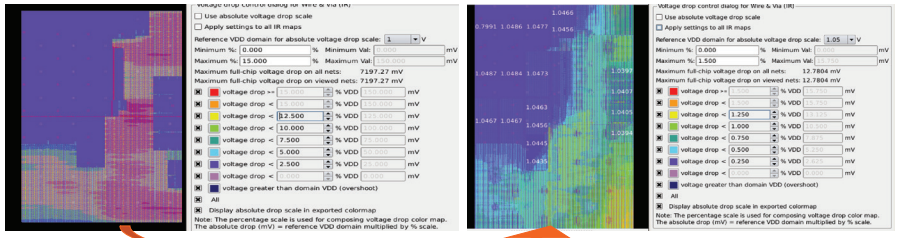
Example of power network with power gating

  arm

PGA enables better power optimization

Power Grid Architect understands Artisan IP architecture

- Create multiple power grids to find optimal solution for power density
- Optimal power networks without wasting routing tracks



  arm

Increasing placement utilization for best area

Run in-depth analysis of all high-level and UNIT level modules

- Placement needs to be in sync with the data flow

Recover area through “reclaimArea” to remove unnecessary buffering

Minimize congestion by customizing the flow for placement uniformity

Optimize the transition times during implementation and signoff

- Ensures the total power balance is maintained

Use “TotalPowerOpt” flow to achieve fine-tuning between the leakage and dynamic power

Review hot spots in timing critical zones using IR aware placement

  arm

Target Clock Tree Synthesis (CTS) for Pmin

POP IP implementation includes many behind the scenes steps to optimize CTS

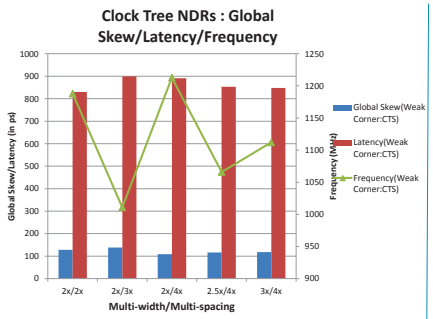
- Evaluation of list(s) of inverters and/or buffers that result in minimized latency, skew and power
- Optimal Clock tree Vt-selection for dynamic power
- Evaluate different structures like Clock Mesh & Flex H Tree for timing critical designs which can compromise a little on dynamic power
- Non-Default rule exploration :
  - Wire rules (width, spacing & layers) that should be used for clocks
  - Shielding techniques (if needed for crosstalk avoidance)

Implementation includes

- Multi-scenario post-CTS optimization for setup, hold and leakage
- Clock tree exceptions to balance timing across sequential stages
- Slack-based hold-fixing at CTS stage to control area bloat

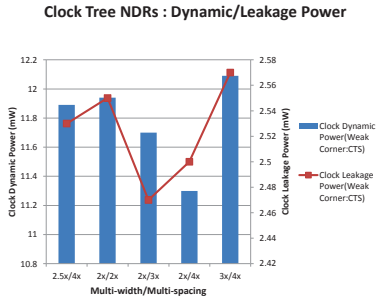
  arm

Minimizing dynamic power through optimal NDRs



2x width / 4x spacing NDR is optimal for timing/skew/latency

2x width / 4x spacing NDR also yields better dynamic power for the clock tree



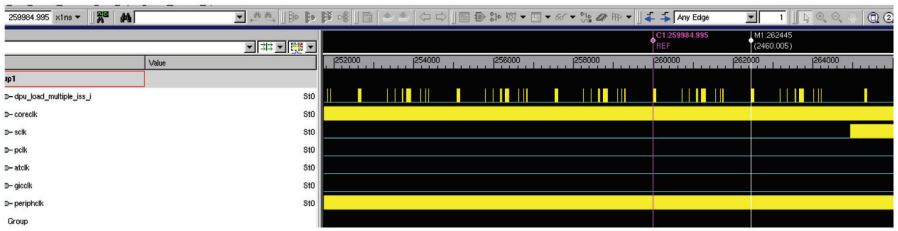
  arm

Investigating dynamic power

Use Power-Indicative Dhrystone vector to report Dynamic / Leakage Power at sign-off

- Measurement flow setup to report power at other vectors also

Loop #4/5 used for power measurements



  arm

Leakage efficiency methodology

Pre-release, POP IP implementation teams explore “forward-fill” Vs “back-fill” techniques

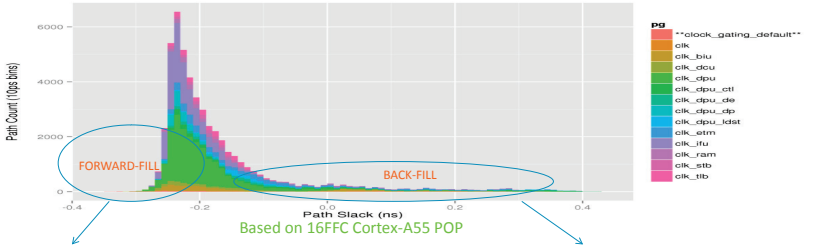
Forward-fill (+Back-fill) :

- Synthesize at mid point Vt/CL for the concerned technology
- Sprinkle some low Vt/CL (targeting the tail) and meet performance
- Back fill to recover leakage with High Vt/CLs

Back-fill :

- Synthesize at low Vt/CL for the concerned technology
- Mainly Edge rate tune only
- Backfill to recover leakage with High Vt/CLs

Leakage efficiency implemented results



Based on 16FFC Cortex-A55 POP

Vt-Usage	Performance	Leakage
100% ULVtC20	2700Mhz	60mW
Forward-fill with ULVtC18 & ULVtC16	2850MHz	65mW
Back-fill with LVt*; SVt*	2850MHz	35mW

Vt-Usage	Performance	Leakage
100% ULVtC16	2850MHz	110mW
Back-fill with LVt*; SVt*;	2850MHz	35mW

Summary

Arm POP IP provide implementation techniques for performance AND area/power optimization

Cortex-A55 POP IP focuses on area and power optimization

- Optimized Artisan physical IP comprising standard cells and memory instances
- Synthesis shmoo for identifying right Vt/CL choices
- Optimal floorplan for best area and power
- Power Grid Architect eases power grid creation enabling optimal solution for power density
- Clock Tree Synthesis targeted for Pmin
- Dynamic and leakage power techniques for power optimization

arm

Thank You